

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant	: Udo Klein et al	Art Unit	: 2174
Serial No.	: 10/675,208	Examiner	: Chris A. Watt
Filed	: September 30, 2003	Conf. No.	: 9931
Title	: VARIABLE SIZE INPUT AREAS IN USER INTERFACES		

Mail Stop Appeal Brief - Patents

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

BRIEF ON APPEAL

(1) Real Party in Interest

SAP Aktiengesellschaft, the assignee of this application, is the real party in interest.

(2) Related Appeals and Interferences

There are no related appeals or interferences.

(3) Status of Claims

Claims 1-20 are pending. All claims stand rejected under 35 U.S.C. § 103.

(4) Status of Amendments

The claims have not been amended subsequent to the final rejection dated May 2, 2007. A Notice of Appeal and a Pre-Appeal Request for Review were filed on August 2, 2007. A decision on the request for review issued on November 23, 2007, which maintained the rejections of all claims. A listing of the current claims is attached.

(5) Summary of Claimed Subject Matter

The claimed subject matter relates to variable size user input areas in computer user interfaces. (Page 1, line 3.) Claims 1-20 are currently pending, and of those, claims 1 and 13 are independent.

CERTIFICATE OF MAILING BY EFS-WEB FILING

I hereby certify that this paper was filed with the Patent and Trademark Office using the EFS-WEB system on this date: January 02, 2008

In the present disclosure, user input areas 202, 206 (FIG. 2) and area 302 (FIG. 3) are examples of user input areas that can be displayed and receive input from a user. (Spec. page 5, line 13—page 9, line 5.) The specification notes that, if characters are represented in a proportional font, a string of a certain number of characters can have different lengths when displayed, because different characters can have different widths. (Spec. 2:3-8.) Such different character widths can cause difficulties or confusion when the user area only accepts up to a fixed number of characters: for example, the specification describes that the area 206 can be limited to ten characters. (Spec. 6:4-5.) This is because the length of an entered text in a proportional font may not indicate whether the entry is close to the maximum allotted number of characters. That is, a string of relatively wide characters may become so long as to give the false impression that few or no additional characters can be entered; similarly, a string of relatively narrow characters may look short and therefore give the false impression that several additional characters can be entered.

The present subject matter provides a solution to these situations by adjusting the input area to a new size. First, the specification describes that the area can be provided with an initial size based on, say, ten times the average width of a character in the proportional font. (Spec. 6:5-7.) Thus, when the area 206 is displayed, its size indicates that the area will accommodate visual representations of the specified number of characters. Independent claims 1 and 13 explicitly recite that there is displayed a user input area that has “a size that visually indicates to a user that the input area will accommodate therein visual representations of the specified number of characters.” In the example of the present disclosure, the user enters the word “Example” and these characters are displayed in the area 206. (Spec. 6:24-25.) Independent claims 1 and 13 explicitly recite that “user input” is received and that the method displays the character(s) in the user input area.

Second, the size of the area 206 is adjusted based on the width of the entered character(s) and on the remainder of the possible data input. (Spec. 6:24-27.) Independent claims 1 and 13 explicitly recite “adjusting the size of the user input area”. The area 206 is then displayed with the “new size”. Independent claims 1 and 13 explicitly state that the new size “visually indicates to the user that the user input area will accommodate therein visual representations of a remaining number of the specified number of characters of the data field”. In the discussed

example of the present disclosure, the area 206 is limited to ten characters and seven of these have so far been entered (i.e., "Example"). Thus, when the average character width is used as a basis, the new width of the area 206 can be equal to the length of the entered characters (Example) plus three characters of average width.

(6) Grounds of Rejection to be Reviewed on Appeal

Claims 1-20 stand rejected as being unpatentable over U.S. Patent No. 5,230,062 (Inaki) in view of U.S. Patent No. 5,450,538 (Glaser) and U.S. Patent No. 6,055,550 (Wallack). Applicants are appealing the rejections of all claims.

(7) Argument

Claims 1-20 are not properly rejected under § 103 because not even the combined disclosures of Inaki, Glaser and Wallack teaches a user input area that is adjusted to a new size as recited in the claims.

Applicants request reversal of this rejection because Inaki, Glaser and Wallack, considered separately or in combination, do not describe or suggest the subject matter of the independent claims 1 and 13, which require that a user input area be displayed with a new size that visually indicates to the user that the user input area will accommodate therein visual representations of a remaining number of the specified number of characters of the data field.

Initially, Applicants note that the final office action characterizes each of Inaki, Glaser and Wallack by assembling selected passages thereof without regard to the context where those statements were made. In the following, Applicants will try to point out some of the instances where this piecemeal interpretation of the references seems to have obscured their true meaning.

Inaki discloses a data processing apparatus and method for defining size and type of data field. Inaki appears to teach that documents generated by a word processing program can be used as the starting point for defining database records. (Inaki 6:7-68.) In a particular example, Inaki describes that the user can place the full-size cursor in the field for "Company Name" (FIG. 11A) and hit a control key to switch to a half-size cursor (FIG. 11B). (Inaki 10:10-66.) Importantly, the user then defines the maximum size of the data input area (e.g., 22 full-size characters) by moving the cursor on the screen (FIG. 11C). (Inaki 10:29-31; 10:40-44.) When

the input field has an acceptable size, the screen can be switched to fill the data input area with dummy characters up to the just-defined limit (FIG. 11D). Thus, the examples of changing the size of an input area in Inaki appear to be directed to a setup phase where the data field is defined, rather than to a use phase where entries are made into the fixed-size fields.

Applicants note that Inaki does not appear to use proportional fonts. Rather, it appears Inaki contemplates characters of standardized widths that can be scaled to either full-size (where the example field holds 22 characters) or half-size (where the field holds 44 characters).

The Examiner cited to Inaki 17:53 et seq. (describing FIG. 24G) as allegedly teaching use of a proportional font. Applicants respectfully disagree. Inaki's FIGS. 24A et seq. show another example where the user is defining the sizes of input fields with the cursor. Thus, when Inaki states "the state of field definition is displayed" as cited by the Examiner, this apparently means that the system has filled in dummy characters up to the maximum field size (see FIG. 24G).

Likewise, the Examiner cited to Inaki 10:31 et seq. (describing FIG. 11C) as allegedly teaching that the user input area has a size that "visually indicates ..." per the present claims. Applicants respectfully disagree. The field that the user defines in Inaki has a capacity of 22 full-size characters or 44 half-size characters. The field, then, has a different capacity depending on which character size is used. Since the input area has a varying capacity, Inaki's input area does not "visually indicate" the capacity of the input area. For example, the FIGS. 11K and 11Q cited by the Examiner lack visual indication for this reason.

The Examiner acknowledged that Inaki does not teach "visual indication of the change in size" or "adjusting the size of the user input area ...". Applicants note that the expressions used by the Examiner are not the exact claim language, and it appears that the Examiner's claim interpretation may not give appropriate (or perhaps any) weight to the requirement in the claims that the new size of the user input area visually indicates that the area will accommodate the "remaining number of the specified number of characters". Nevertheless, Applicants agree that Inaki fails to disclose an input field that is adjusted to a new size after user input, wherein the new size visually indicates that the area will accommodate the remaining number of characters.

The Examiner then cited to Glaser and Wallack to provide the subject matter missing from Inaki. As best understood from the advisory action, the Examiner's argumentation appears to go along the lines of: Inaki discloses input areas but does not visually indicate a change in

size. The Examiner then contends that Glaser teaches such a visual indication of a change. However, the Examiner acknowledges that the change in Glaser is not based on the content of the field, and therefore cites Wallack as allegedly showing field adjustment based on content.

Applicants respectfully disagree with the position that even the combined disclosure of Inaki, Glaser and Wallack teaches the present subject matter or renders it unpatentable. Moreover, the combination of these references appears to be based entirely on hindsight from the subject matter in the present disclosure. Therefore, and as articulated below, Applicants assert that a person of ordinary skill in the art would not have combined the references as alleged by the Examiner.

Glaser discloses a graphical user interface control for expansion and re-sizing of data fields in forms. Particularly, Glaser uses a small darkened rectangle 142 (FIGS. 2, 4) which the user can grab and drag to expand or contract a data entry field. (Glaser 5:29—6:1.) Like Inaki, however, Glaser does not appear to describe proportional fonts. Moreover, Glaser's data entry field can apparently be resized independently of the maximum number of characters of the data field, and independently of any entry made in the field. Therefore, Glaser also does not perform the visual indication required by the present claims.

Wallack, finally, discloses automatic sizing of fields for displaying computer forms. Wallack appears to teach that based on the size of a sample record a system can decide the column width to use for displaying a number of records. (Wallack 3:11—4:47.) In short, after a user selects a column to resize, the system retrieves a sample record and reads the size of the entry for the selected column. (Wallack 4:48-50.) The system then resizes the column to fit the largest amount of data found in the sample record(s). Accordingly, the size of the column cell is chosen so that an entry that has already been made will fit without obstruction. Wallack acknowledges that when the column size is set based on a sample, any record that contains more data in the field than the sample may require scrolling. (Wallack 4:61-63.) Like Inaki and Glaser, however, Glaser does not appear to describe proportional fonts. There is also no disclosure of taking into account a maximum number of characters that the field will accept into the resizing. As such, the fields taught by Wallack fail to visually indicate that the resized field will accommodate a remaining number of characters.

Applicants submit that there is no suggestion for combining the references as asserted in the office action. The purported reason is "to auto size the field according to textual data received in the fields in a computer generated form". This reasoning appears to be entirely based on hindsight, because such a combination of the references appears to substantially or entirely contradict the explicit teachings in the references.

First, Glaser teaches that the user can selectively decide the size of the input field by dragging its borders, while Wallack teaches automatic resizing based on sample contents. Applicants submit that implementing Wallack's automatic resizing into Glaser would completely eradicate the user's freedom to adjust the field size. As such, there does not seem to exist any suggestion or motivation to combine the references.

Second, even if Glaser were modified by Wallack's automatic resizing as suggested, this would defeat a feature explicitly touted in Inaki, namely that the user can define the size of the input field using the cursor (e.g., to 22 full-size characters). That is, based on the sample chosen according to Wallack the fields of Inaki may be defined to include, say, 22 full-size characters. However, the 22-character definition in Inaki restricts the capacity of the data filed itself, not merely how it is displayed to the user. Other records that were not sampled may require more data in that field (indeed, Wallack acknowledges that some records may contain more data than the sample). Thus, allowing the data size of a sample record (as taught by Wallack) be the sole determining factor for defining the number of characters allowed in the field (which is how Inaki works) appears to significantly reduce the usefulness of Inaki.

For at least these reasons, Applicants respectfully submit that the rejection of independent claims 1 and 13 is improper. Claims 2-12 and 14-20 contain the respective limitations of the independent claims due to their dependency, and also recite additional language that further specifies some of the features discussed above. As the references do not teach several features of the independent claims per the discussion above, the dependent claim features also do not appear to be present in the references. Accordingly, Applicants believe that all pending claims 1-20 are patentable over the references of record.

Conclusion

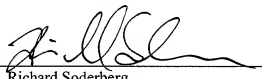
Accordingly, for at least the above reasons, Applicants request that the Board overturn the rejections of the pending claims.

A Notice of Appeal and Pre-Appeal Brief Request for Review was filed on August 2, 2007. A petition for extension of time for three months is being filed with this Appeal Brief.

The extension fee in the amount of \$1050 and \$500 in payment of the brief fee is being paid concurrently herewith on the Electronic Filing System (EFS) by way of Deposit Account authorization. Please apply any other charges or credits to Deposit Account No. 06 1050.

Respectfully submitted,

Date: 1/2/08


J. Richard Soderberg
Reg. No. 43,352

Customer No.: 32864
Fish & Richardson P.C.
3300 Dain Rauscher Plaza
60 South Sixth Street
Minneapolis, MN 55402
Phone: (612) 335-5070
Fax: (612) 288-9696

Appendix of Claims

1. A method of displaying a user input area, the method comprising:

displaying a user input area within a computer user interface, wherein the user input area corresponds to a data field having a specified number of characters and has a size that visually indicates to a user that the user input area will accommodate therein visual representations of the specified number of characters of the data field;

upon receipt of a user input specifying a character to be included in the data field, displaying within the user input area a visual representation of the input character in a proportional font;

adjusting the size of the user input area based on a size of characters included in the data field and the specified number of characters of the data field, wherein the size of characters included in the data field includes a size of the input character; and

displaying the adjusted user input area having a new size that visually indicates to the user that the user input area will accommodate therein visual representations of a remaining number of the specified number of characters of the data field.

2. The method of claim 1, wherein the user input area is displayed only when the user input area has focus.

3. The method of claim 1, wherein the user input area contains a character before the user input specifying the character is received.

4. The method of claim 1, wherein the user input area is empty when the input specifying the character is received, and wherein the user input area size then is equal to the specified number of characters times a selected character width.

5. The method of claim 4, wherein the selected character width is an average width of characters.

6. The method of claim 1, wherein the size of the user input area after the specified character is displayed equals the width of the displayed character plus the remaining number of the specified number of characters times a selected character width.

7. The method of claim 1, wherein the size of the user input area is adjusted after each character that is received.

8. The method of claim 1, further comprising adjusting the size of the user input area differently after receiving a second last character of the specified number of characters.

9. The method of claim 8, further comprising adjusting the user input area, after receiving the second last character, to equal a cumulative width of all characters displayed in the user input area plus a selected character width.

10. The method of claim 9, wherein the selected character width is a maximum width of characters.

11. The method of claim 1, further comprising adjusting the size of the user input area after receiving the specified number of characters, to equal a cumulative width of the characters displayed in the user input area.

12. The method of claim 1, wherein a user input specifying a character to be removed from the data field is received, further comprising displaying the user input area without the removed character, the user input area having a size equal to a cumulative width of any characters displayed in the user input area plus the remaining number of the specified number of characters times a selected character width.

13. A computer program product containing executable instructions for displaying a user input area within a computer user interface, the instructions when executed causing a processor to:

display the user input area within the computer user interface, wherein the user input area corresponds to a data field having a specified number of characters and has a size that visually indicates to a user that the user input area will accommodate therein visual representations of the specified number of characters of the data field;

upon receipt of a user input specifying a character to be included in the data field, display within the user input area a visual representation of the input character in a proportional font;

adjust the size of the user input area based on a size of characters included in the data field and the specified number of characters of the data field, wherein the size of characters included in the data field includes a size of the input character; and

display the adjusted user input area having a new size that visually indicates to the user that the user input area will accommodate therein visual representations of a remaining number of the specified number of characters of the data field.

14. The computer program product of claim 13, wherein the size of the user input area after displaying the input character equals the width of the character plus the remaining number of the specified number of characters times a selected character width.

15. The computer program product of claim 13, wherein the remaining number of the specified number of characters is received in the user input area, further comprising instructions that when executed cause the processor to:

display the user input area with a size equal to a cumulative width of the displayed specified number of characters in the user input area.

16. The computer program product of claim 13, further comprising instructions that when executed cause the processor to:

adjust the size of the user input area differently after receiving a second last character of the specified number of characters.

17. The computer program product of claim 16, further comprising instructions that when executed cause the processor to:

adjust the user input area, after receiving the second last character, to a size that is equal to a width of all characters displayed in the user input area plus a selected character width.

18. The computer program product of claim 17, wherein the selected character width is a maximum width of characters.

19. The computer program product of claim 13, wherein the new size that visually indicates to the user that the user input area will accommodate therein visual representations of a remaining number of the specified number of characters is a different size than the size that visually indicates to a user that the user input area will accommodate therein visual representations of the specified number of characters.

20. The computer program product of claim 13, wherein the new size that visually indicates to the user that the user input area will accommodate therein visual representations of a remaining number of the specified number of characters is the same size as the size that visually indicates to a user that the user input area will accommodate therein visual representations of the specified number of characters.

Applicant : Udo Klein et al
Serial No. : 10/675,208
Filed : September 30, 2003
Page : 12 of 13

Attorney's Docket No.: 16104-005001 / 2003P00582 US

Evidence Appendix

None.

Applicant : Udo Klein et al
Serial No. : 10/675,208
Filed : September 30, 2003
Page : 13 of 13

Attorney's Docket No.: 16104-005001 / 2003P00582 US

Related Proceedings Appendix

None.